**Dokumentation SBB Fahrplan**







Inhaltsverzeichnis

[Einleitung 3](#_Toc27732952)

[Anforderungen 3](#_Toc27732953)

[Mockup 4](#_Toc27732954)

[Use Case 5](#_Toc27732955)

[Use Case Beschreibung 6](#_Toc27732956)

[Aktivitätsdiagramm 8](#_Toc27732957)

[Programierrichtlinen 9](#_Toc27732958)

[Testprotokoll 11](#_Toc27732959)

# Einleitung

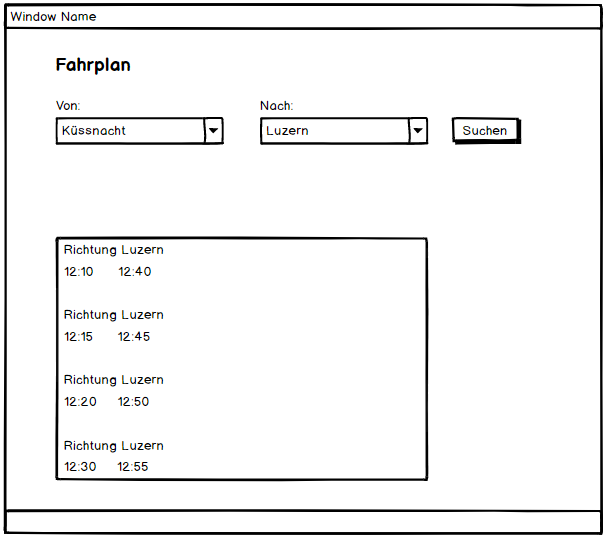
Im ÜK Modul 318 Analysieren und objektbasiert programmieren ist das Projekt die Realisierung eines Fahrplans. Dies wird im Visual Studio programmiert und umgesetzt mit Windows Forms. Der Fahrplan ermöglicht die Eingabe von zwei Standorten inklusiven Datums und Uhrzeit Eingabe. Das Programm zeigt anschliessen 4-5 Verbindungen, die der User für die perfekte Verbindung benutzen kann.

# Anforderungen

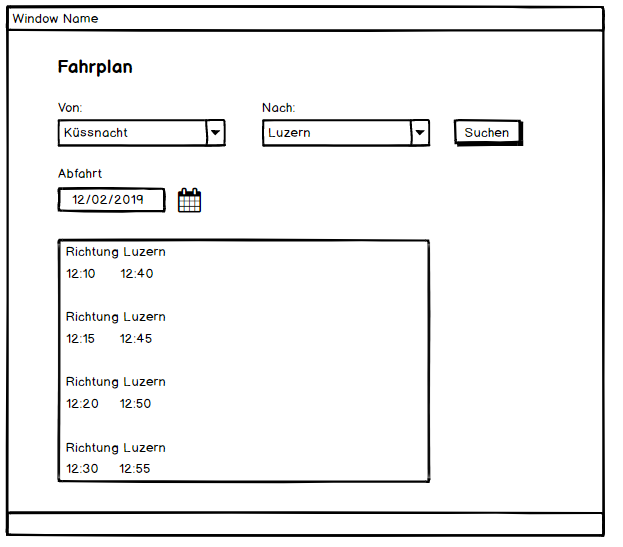
|  |  |  |
| --- | --- | --- |
| ID | Beschreibung | Erfüllt |
| **A001** | Als ÖV-Benutzer möchte ich Start- und Endstation mittels Textsuche suchen können, damit ich nicht alle Stationen auswendig lernen muss. | Ja |
| **A002** | Als ÖV-Benutzer ich die aktuellen, d.h. mindestens die nächsten 4 bis 5 Verbindungen zwischen den beiden gefundenen und ausgewählten Stationen sehen, damit ich weiss wann ich zur Station muss, um de für mich idealen Anschluss zu erwischen | Ja |
| **A003** | Als ÖV-Benutzer möchte ich sehen, welche Verbindungen ab einer bestimmten Station vorhanden sind, damit ich bei mir zuhause eine Abfahrtstafel haben kann. | Ja |
| **A004** | Als ÖV-Benutzer möchte ich, das schon während meiner Eingabe erste Such-Resultate erscheinen, damit ich effizienter nach Stationen suchen kann. | Ja |
| **A005** | Als ÖV-Benutzer möchte ich nicht zur aktuellen Verbindung suchen können, sondern auch solche zu einem beliebigen anderen Zeitpunkt, damit ich zukünftige Reisen planen kann. | Ja |
| **A006** | Als ÖV-Benutzer möchte ich sehen, wo sich eine Station befindet, damit ich mir besser vorstellen kann, wie die Situation vor Ort aussieht. | Nein |
| **A007** | Als ÖV-Benutzer möchte Stationen finden, die sich ganz in der Nähe meiner aktuellen Position befinden, damit ich schnell einen Anschluss erreichen kann. | Nein |
| **A008** | Ich möchte meine gefundenen Resultate via Mail weiterleiten können, damit auch andere von meinen Recherchen profitieren können. | Nein |

# Mockup

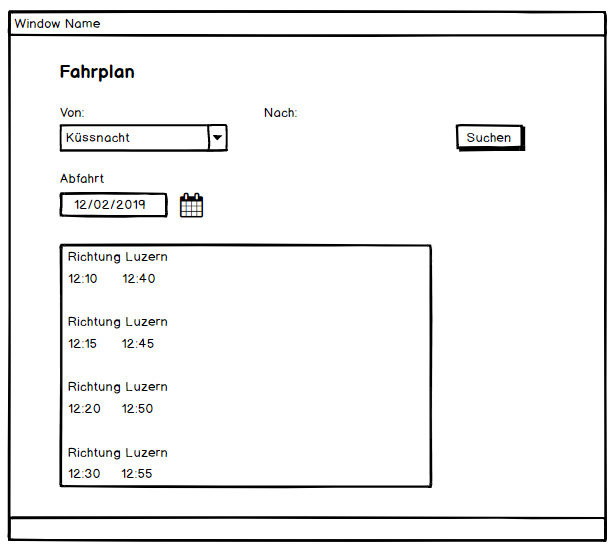
Eine grobe Übersicht vom GUI wurde im Balsamiq gezeichnet.



A001: Start und Endstation mittels Textsuche suchen können.

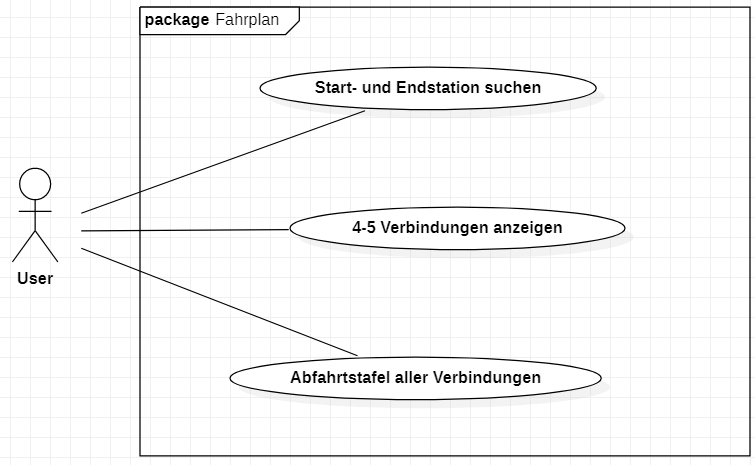


A002: Aktuell Verbindungen der nächsten 4-5 Verbindungen anzeigen.



A003: Verbindungen ab einer bestimmten Station anzeigen.

# Use Case



Das Use Case für die Anforderung A001 bis A003 wurde im StarUML gezeichnet.

# Use Case Beschreibung

A001

|  |  |
| --- | --- |
| **Use Case** | **1, Verbindung suchen** |
|  |  |
| **Beschreibung** | Start- und Endstation mittels Textsuche suchen |
| **Akteure** | Ein User der das Programm bedient. |
| **Auslöser** | Eine Abfrage |
| **Vorbedingungen** | Der User muss das Programm installiert haben |
| **Ablauf** | 1. Programm öffnen 2.Abfahrt- und Zielort eingeben 3.Suchen klicken |
| **Alternativer Ablauf** | kein |
| **Ergebnisse** | Anzeige des Fahrplans |

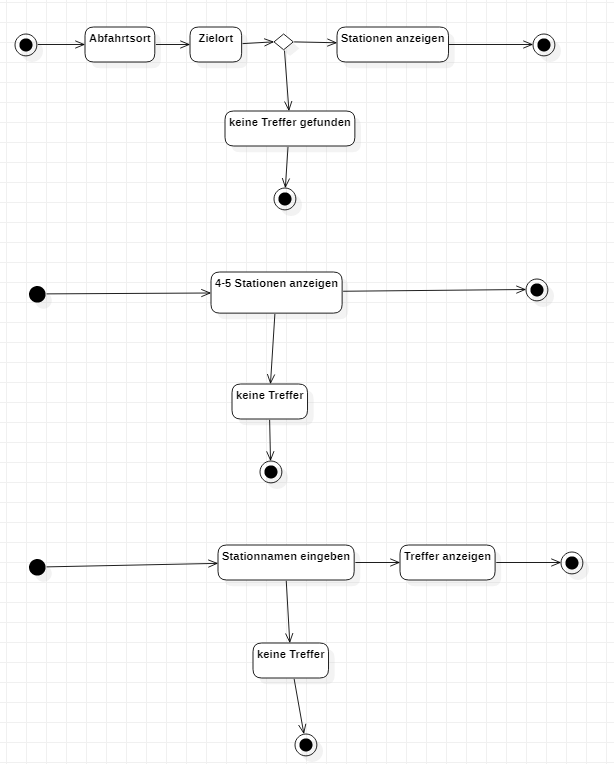
A002

|  |  |
| --- | --- |
| **Use Case** | **2, 4-5 Verbindungen anzeigen** |
|  |  |
| **Beschreibung** | Der Benutzer möchte die nächsten 4-5 Verbindungen anzeigen |
| **Akteure** | Ein User der das Programm bedient. |
| **Auslöser** | Eine Abfrage |
| **Vorbedingungen** | Der User muss das Programm installiert haben |
| **Ablauf** | 1. Programm öffnen 2.Abfahrt- und Zielort eingeben 3. Aktuelle Uhrzeit und Datum auswählen 4.Suchen klicken |
| **Alternativer Ablauf** | kein |
| **Ergebnisse** | Anzeige des Fahrplans |

A003

|  |  |
| --- | --- |
| **Use Case** | **3, Aktuelle Verbindungen sehen** |
|  |  |
| **Beschreibung** | Der Benutzer möchte alle Verbindungen von einer bestimmten Station sehen |
| **Akteure** | Ein User der das Programm bedient. |
| **Auslöser** | Eine Abfrage |
| **Vorbedingungen** | Der User muss das Programm installiert haben |
| **Ablauf** | 1. Programm öffnen 2.Abfahrt- und Zielort eingeben 3. Aktuelle Uhrzeit und Datum auswählen 4.Suchen klicken |
| **Alternativer Ablauf** | kein |
| **Ergebnisse** | Anzeige des Fahrplans |

# Aktivitätsdiagramm



A001

A002

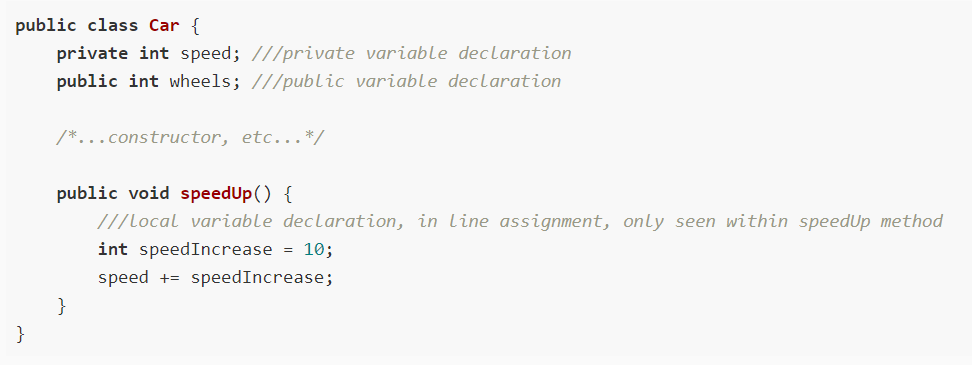
A003

Das Aktivitätsprogramm für A001, A002 und A003 diese wurden ebenfalls im StarUML gezeichnet.

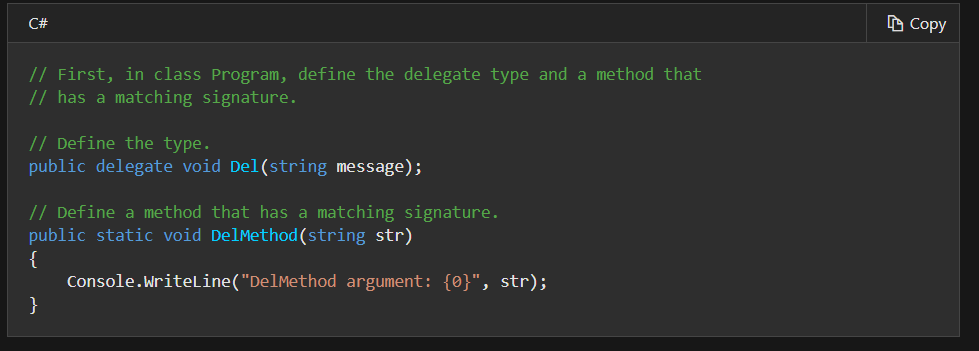
# Programierrichtlinen

Naming Conventions

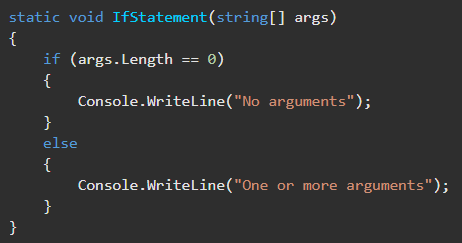
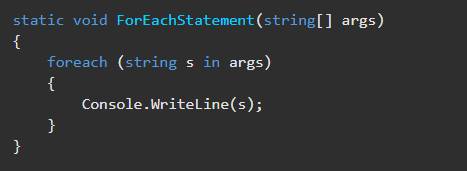
* Soll immer auf Englisch sein, ausser für den Anwender sollte auf Deutsch sein
* Generell PascalCase verwenden.
* Constants in UPPER\_CASEDeclaration

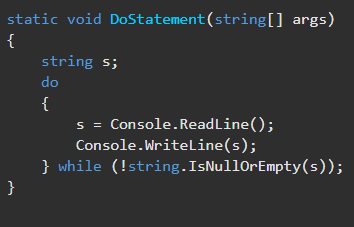
Beispiel

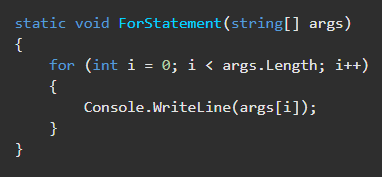
**Comments**

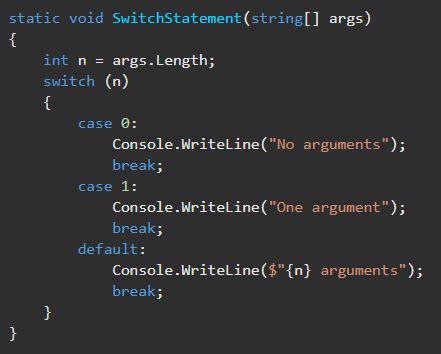
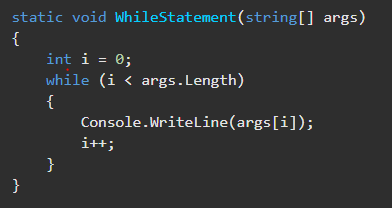
Kommentare können mit //, /\* \*/ und # geschrieben werden.

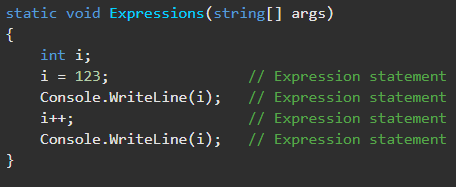
**Statements (Beispiele)**

****if else foreach

**** do while

for

****switch case while

**** int

# Testprotokoll

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Schritt** | **Aktivität** | **Erwartetes Resultat** | **Tatsächliches Resultat** | **Status** |
| 1 | Der User füllt die "Angaben" bei der "Verbindungssuche" aus "Von: Nach:" und klick anschliessend auf Suchen | Die Daten Werden Angezeigt "Datum, Abfahrtszeit, usw. | Die Daten Werden Angezeigt "Datum, Abfahrtszeit, usw. |  |
| 2 | Der User füllt zusätzlich die "Erweiterte Angaben", "Datum" und "Zeit" aus und klick anschliessend auf Suchen | Die Daten Werden Angezeigt "Datum, Abfahrtszeit, usw. | Die Daten Werden Angezeigt "Datum, Abfahrtszeit, usw. |  |
| 3 | Der User füllt den "Abfahrtsort" bei der "Abfahrtsübersicht" aus und klick anschliessend auf Suchen | Die Daten "Abfahrt", "Richtung" und "Linie" werden angezeigt. | Die Daten "Abfahrt", "Richtung" und "Linie" werden angezeigt. |  |